AMENDMENTS TO THE CLAIMS

Claims 1-33 are pending in the Application, and all have been rejected in the Office action mailed November 20, 2006. Claims 1, 9, 11, 15, 18, 19 and 20 have been amended in this response. Claims 1 and 18 are independent claims. Claims 2-17 and 19-33 depend, respectively, from independent claims 1 and 18.

The following listing of claims replaces all previous versions, and listings, of claims in the Application.

Listing of Claims:

1. (Currently amended) A system for generating efficient and compact update packages for updating contents of <u>memory in</u> an electronic device utilizing source and target images of the contents, the system comprising:

<u>at least one processor communicatively coupled to storage containing code</u>

<u>executable by the at least one processor, the code comprising:</u>

a parser for generating distance files <u>comprising distance information</u>

<u>representing location differences</u> between <u>code or objects in</u> the source image

and the target image;

a bubble generator for generating ~~bubbles~~ <u>bubble information from the</u>

<u>distance information, the bubble information representing addition and deletion of</u>

<u>memory space within the source image to more closely align the code or objects</u>

<u>in the source and target images</u>;

a configuration manager for facilitating configuration of memory elements

of the electronic device;

a bubble layout manager for modifying the source image to look similar to

the target image<u>, based upon the bubble information</u>; ~~and,~~ <u>and</u>

a generator for generating <u>at least one</u> update ~~packages~~ <u>package from the</u> <u>modified source image and the target image, for processing in the electronic</u> <u>device to update the memory</u>.

2. (Original) The system according to claim 1 wherein the parser preprocesses map files for generating the distance files.

3. (Original) The system according to claim 1 wherein the parser preprocesses symbol files for generating the distance files.

4. (Original) The system according to claim 1 wherein the bubble generator processes the distance files to generate a list of bubbles.

5. (Original) The system according to claim 1 wherein the bubble generator outputs a file containing a list of the generated bubbles.

6. (Original) The system according to claim 1 wherein the parser generates a plurality of distance files associated with a plurality of memory components in the electronic device.

7. (Original) The system according to claim 6 wherein the bubble generator processes the plurality of distance files to generate a plurality of corresponding files containing bubbles information.

8. (Original) The system according to claim 7 wherein the bubble generator utilizes the plurality of files containing bubbles information for generating an output file containing a portion of the generated bubbles.

9. (Currently amended) The system according to claim 1 wherein the bubble layout manager comprises:
a bubbler; ~~and,~~ <u>and</u>

a predictor for aligning objects between the source and target images.

10. (Original) The system according to claim 1 wherein the update package comprises a set of instructions and data.

11. (Currently amended) The system according to claim 10 wherein the generator comprises:

a residue processing unit for minimizing the number of instructions in the update package; ~~and,~~ and

an update package output for generating the update package.

12. (Original) The system according to claim 1 wherein the generator determines an appropriate bank order of updates.

13. (Original) The system according to claim 12 wherein the appropriate bank order provides a more size-efficient update package.

14. (Original) The system according to claim 12 wherein the appropriate bank order provides a lesser number of instructions in the update package.

15. (Currently amended) The system according to claim 1 wherein the system further comprises:

an entropy calculator for calculating the entropy of a segment of data; and

~~and,~~ a compression unit for facilitating compression of the update package.

16. (Original) The system according to claim 15 wherein the residue processing unit utilizes the calculated entropy to select a set of instructions to determine the update package.

17. (Original) The system according to claim 16 wherein the entropy is calculated for different sets of instructions to determine the instruction set yielding the smallest entropy value.

18. (Currently amended) A method for generating efficient and compact update packages ~~in a generation system having a parser, a bubble generator, a configuration manager, a bubble layout manager, and a generator, the update packages~~ for updating contents of <u>memory in</u> an electronic device, utilizing ~~the~~ source and target images of the contents, the method comprising the steps of:

identifying ~~determining~~ files <u>comprising code or objects of</u> ~~for~~ the source image;

identifying ~~determining~~ files <u>comprising code or objects of</u> ~~for~~ the target image;

creating <u>one or more</u> distance files for the source and the target images<u>, the one or more distance files comprising information representing differences of location of the code or objects in the source and target images</u>;

generating bubble information <u>using the one or more distance files, the bubble information representative of addition and deletion of memory space within the source image</u>;

applying the bubble information to the source image <u>to create a modified source image in which the code or objects more closely align with corresponding code or objects in the target image</u>;

generating an update package <u>using the modified source image and the target image</u>; and

outputting the update package and the bubble information <u>to the electronic device for processing to update the memory</u>.

19. (Currently amended) The method according to claim 18 wherein the ~~determined~~ <u>identified</u> files for the source and the target images are parsed to create the distance files.

20.   (Currently amended)   The method according to claim 18 wherein the distance files are split into at least ~~one part~~ two parts.

21.   (Original) The method according to claim 18 further comprising the step of verifying the distances between the source image and the target image.

22.   (Original) The method according to claim 18 wherein the bubble information is configured according to configuration settings.

23.   (Original) The method according to claim 18 wherein the distance files are split into parts corresponding to different sections of memory.

24.   (Original) The method according to claim 20 wherein the distance files are split into parts corresponding to different sections of code.

25.   (Original)   The method according to claim 22 wherein the configuration settings are specified by a user.

26.   (Original)   The method according to claim 22 wherein the configuration settings are default configuration settings.

27.   (Original) The method according to claim 18 wherein the application of the bubble information to the source image is performed by a bubbler.

28.   (Original) The method according to claim 27 wherein the bubbler utilizes a predictor.

29.   (Original) The method according to claim 18 wherein the generation of the update package utilizes residue processing.

30.   (Original) The method according to claim 18 wherein the generation of the update package utilizes compression.

31.   (Original) The method according to claim 18 wherein the update package and the bubble information are packaged together and outputted.

32. (Original) The method according to claim 18 wherein the update package and the bubble information are packaged and outputted separately.

33. (Original) The method according to claim 18 wherein the update package and the bubble information are packaged together and saved in a file.